

# Configuration of Redundant Group (Server)

## Configuration of redundant group (Server)

Redundant group (RDG) is formed by one or several application servers. Each of them is installed on a different computer. When starting the application, the server tries to read RDS parameters uniquely bound with the application.

Parameter	Meaning
GroupName	Text string defining the RDG name. If the server doesn't find the parameter on start-up, or it is a string with zero length, the attempt to insert the server into the RDG will not be executed and the application will run with no redundancy support. The parameter allows to disable all redundancy features before starting the server and to run the application in normal mode.
KernelName	Unique name of application server within the RDG. If the parameter doesn't exist, or is an empty text, there will be used the computer name (Host Name).
State	Required state of the application server after starting. In current implementation there's only one allowed state - SBS.
Priority	Priority of the application server in regard to the others included in the RDS. Higher number means higher priority. The priority is used for unexpected failure of the HS and defines which SBS takes over the HS functionality (becomes the HS). The priority of 0 disables the automatic changing the server status into the status HS. This is allowed by means of the process <a href="#">D2000 System Console</a> .

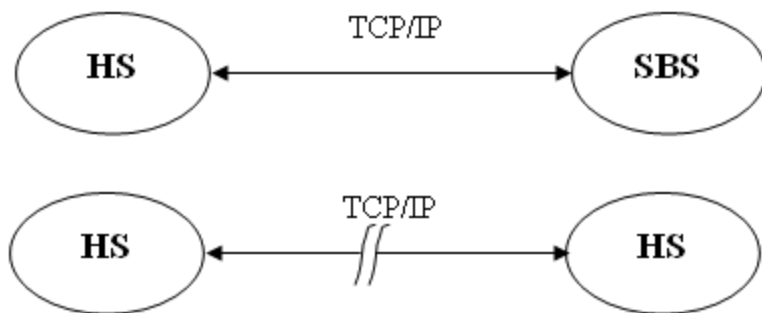
Application server inserted into a RDS can get the following states:

State	Value	Description
HS	0	Active server within RDS
SBS	1	STANDBY server
CS	2	Crashed server
SS	3	Server is starting
FS	4	Wrong setting of parameter <a href="#">State</a>
TS	5	Test server: not implemented

Changes of the server states described in the table above are also characterized by changes into temporal states, which are specified by RDS parameters. The parameters are described in the following table:

Temporal state	RDS parameter limiting the state [s]	Description
iNone	RD_TIMEOUT_iNone	Stable state
iElection	RD_TIMEOUT_iElection	Election
iWaitingHot	RD_TIMEOUT_iWaitingHot	Waiting for <b>HS</b>
iWaitingReadyHot	RD_TIMEOUT_iWaitingReadyHot	Waiting for ready <b>HS</b>
iStartingKernelToSBS	RD_TIMEOUT_iStartingKernelToSBS	Starting the server to <b>SBS</b> state
iStartingKernelToHot	RD_TIMEOUT_iStartingKernelToHOT	Starting the server to <b>HS</b> state
iHotOrSBSToSBS_WaitForHot	RD_TIMEOUT_iHotOrSBSToSBS_WaitForHot	Waiting for <b>HS</b> after required change
iHotOrSBSToSBS_WaitAnsConn	RD_TIMEOUT_iHotOrSBSToSBS_WaitAnsConn	Waiting for confirmation of <b>SBS</b> log on to <b>HS</b>

In regard to the fact, that RDG contains several running application servers, which are ready to take over the HS function if the HS fails, it is of highets important to ensure that several SBS servers will not change their states into the state HS when some communication routes fail. Such a situation can occur, when the communication card of the computer with a running RDG member in the state SBS fails. From the RDG member's point of view - the HS has failed and it is trying to replace the HS and get to the state HS.



To prevent the state described above from occurring, each application server, inserted into RDG, test the "visibility" of at least IP address from given list using the ICMP protocol by means of the service PING. PING is not successful, if the service is terminated by an error or is not terminated within given time limit. If none of the addresses is visible, the status of the application server is changed into the state SC and is terminated.

List of IP addresses and time limit are part of RDS parameters:

Parameter	Description
NetCheck_Ping_TIME_OUT	Time limitation for the PING service [ms]
NetCheck_Ping1	STANDBY server
NetCheck_Ping2	IP address
....	....
NetCheck_PingN	IP address

In real application, it is appropriate if each RDG member checks for presence of the other members and at least one computer, that is not a RDG member. Consequence of the activity is :

1. Server refuses to run (and then changes its status into the state HS) if all computers included in the PING service table are disabled (or unavailable to the PING service).

When you set the constant *NetCheck\_Ping\_TIME\_OUT* it is important, if the computer, which is not connected to the network detects the fact before finishing of the state iElection. If no address from the list is available, the server checks the list again before changing its status into the state CS. So in the extreme, it checks the list twice. This operation may take the time of  $2 \cdot N \cdot \text{NetCheck\_Ping\_TIME\_OUT}$ . N is the number of NetCheck\_Ping addresses. The time must be lower that *RD\_TIMEOUT\_iElection*.

$2 \cdot N \cdot \text{NetCheck\_Ping\_TIME\_OUT} < \text{RD\_TIMEOUT\_iElection}$   
and therefore  
 $\text{NetCheck\_Ping\_TIME\_OUT} < \text{RD\_TIMEOUT\_iElection} / (2 \cdot N)$

For example: If  $N=6$  a  $\text{RD\_TIMEOUT\_iElection} = 7$  [s], there must be valid:

$\text{NetCheck\_Ping\_TIME\_OUT} < 7\,000 / (2 \cdot 6)$   $\text{NetCheck\_Ping\_TIME\_OUT} < 580$  [ms]

The whole information interchange among RDG members is executed by means of **MULTICASTS**. This specifies the limitation of the set of computers, where RDG members can be placed on. For the proper functionality it is required, that the server as a RDG member must recognize the following parameters:

Parameter	Description
IPMask	IP mask of the network, where the addresses IPAddr1 and IPAddr2 belong to
IPAddr1	Server IP address on primary network
IPAddr2	Server IP address on secondary network

Location of these parameters is described in the chapter [Location of configuration parameters](#).

IPAddr1 is the IP address of the server to which the clients will connect. If a secondary communication network is used for reasons of safety and redundancy, parameter IPAddr2 should be defined too.

If neither of parameters IPAddr1 and IPAddr2 is defined (or they're both empty strings), server running on Windows platform will query the IP addresses from operating system. For OpenVMS platform parameters IPAddr1 and IPAddr2 are mandatory if the server is part of a redundant group.

Note: if the server has more than 2 interfaces or more than 2 IP addresses (e.g. IP aliases), it is recommended to set the parameters IPAddr1 and IPAddr2, because dynamic detection of addresses doesn't guarantee the order of IP addresses in which the operating system will provide them (the first two obtained IP addresses are used, loopback address 127.0.0.1 is not taken into account).

Manual setting of parameters IPAddr1 and IPAddr2 is especially recommended in systems with changing IP addresses (dynamic aliases, clusters etc) as D2000 Server only queries IP addresses during startup.

The addresses are broadcasted within the network by means of MULTICASTS, when querying on the RDG state. Queries are used by individual application servers inserted into the RDG as well as by clients connected to the RDG when using the parameter [/RD](#).

**Related pages:**

[Application in redundant system](#)

[Location of configuration parameters](#)

[Temporal server states and RD\\_TIMEOUT parameters](#)

[Synchronization of configuration databases](#)